Project Guideline

DSCI 551, Spring 2021

The theme of the project is to apply data management techniques to solve an interesting real-world problem. The focus is on the data management aspect of the problem. In particular, think about where the data come from (data collection), how they should be stored (data modeling), which databases are suitable for storing them (SQL or NoSQL) to facilitate query processing, and how to scale it up when there are large amount of data (parallel processing, e.g., Hadoop & Spark). Furthermore, for multi-person team, we are requiring you to also address the situation where data sets are not fixed, instead continuously arriving, e.g., event data, log data, sensor data, stock, weather, etc. Here are detailed requirements.

- Data collection: you need to create at least one dataset by yourself. Such a dataset should not already be available from the Web. There are a number of ways of obtaining such data. For example, utilize public API of web sites. For example,
    - Twitter API
    - Flickr API
    - NOAA API
    - Yelp Fusion API

  For another example, you can collect the server logs on your machine, e.g., EC2 instance (Hadoop, Spark logs, etc.). Please be creative on the problems and data sets. For example, connect it to your backgrounds and interests. It is a good idea to share the APIs you like to work on in the project brainstorming sheet or Piazza, including details on how to work with API.
- Databases: you are required to use at least one database to store your data set. Depending on the structure of your data, provide a reason that why you select one database over another. We encourage you to explore new databases not covered in class as said. For example, Cassandra, HBase, Redis, Hive, Pig, etc.
- Parallel processing: you are required to develop a solution using either Hadoop or Spark or Hive to process data in parallel. The processing may be for some kinds of analysis on the data set that provide insights.
- UI: develop an intuitive UI for the solution.
- **Data science challenge**: in about two weeks after the proposal is due, you will be asked to prepare your datasets and problems into a data science challenge (refer to Kaggle for some examples). Multi-person teams will be required to also tackle the challenges posed by other teams (details below).

You can form a team of up to three people. However, multi-person teams need to satisfy the following additional requirements.

- 2-person teams:
    - Work on at least one data science challenge (we will ask you to submit preferences)

    - Preferably at least one data set is streaming data (e.g., you may turn tweets and logs into continuous data source) and you need to use stream processing engine (e.g., Spark and Kafka)
- 3-person teams: satisfy the following requirements in addition to that for 2-person teams:
    - At least one team member should come from Non-CS/IT background
    - At least two data science challenges
    - At least one NoSQL database not covered in class
    - At least one aspect of parallel data processing not covered in class (Spark GraphX is ok).
    - At least one dataset is streaming data.

Note that forming a team with people from different sections is not permitted.

We will conduct peer evaluation on the data science challenges you created and your overall project deliverable at the end of semester. In addition, we will be asking you to create a video to present the aspects of your project in depth, at the end of semester.

Your "data science challenge" will be evaluated based on: whether the problem is interesting, the problem is clearly defined, data set is clearly described and adequate to address the problem, and data set is in a format amiable to be adopted by others, etc. More detailed guideline will be posted when the time is approaching.

Your final project will be evaluated based on the interestingness of problems, data sets, solutions, UI friendliness, and any Wow factors in the project, etc.

Project grading:

- Proposal & presentation: 10% (2% for presentation)
- Data science challenge (creation): 5%
- Midterm report: 10%
- Implementation: 50%
- Final report: 10%
- Final demo and presentation: 10% (2% for video)
- Peer evaluation: 5%

Proposal requirements:

- Team members and their undergraduate majors and experiences.
- Reason on the team formation
- Description of the problem you are solving
- The datasets you are creating. Provide a link to any web site.
- Sample data from your data set
- How you plan to approach the problem.

Midterm progress:

- Your data sets and problems should have been well created and defined.
- Data collection should be done at this point.

- Data should be processed and stored in the database you selected.
- State other progresses on problem solving, parallel (and stream for multi-person team) processing, and UI creation.

Final report:

- It should be accumulative: problem description & solution
- Reflections on experiences and lessons learned.
- Multi-person teams: also report on the solution to the data science challenges from peers.

Final video: record about 10-20 minutes video on the project, e.g., using Zoom recording (with camera on). All persons in the group should be present in the presentation.

Research project opportunity: This requires you to already be familiar with MongoDB, Spark DataFrame, and MySQL. The goal of the project is to create query translation tool to facilitate students in their learning of the above mentioned systems. It takes query user formulated using HTML form, translate it to MongoDB operation, etc. and also support live execution of translated queries. Works by students from past semester (video: Query translation) may be taken as a starting point. The tool should be available on the Web.

If you are interested, please talk to me directly. Note that the project requires you to have solid programming skills on UI development. Ideally, the tool at least part of it can be available to the students after midterm (in time for learning MongoDB and Spark).

Note that you are also required to meet the data collection part of the general requirements.